

## **REMARKS**

Applicant respectfully requests reconsideration and allowance of the subject application. Claims 1-12 and 23-37 were previously withdrawn and are canceled without prejudice as part of this Response. Claims 1-12 and 23-37 are being pursued in one or more divisional applications. New claim 47 is added. Claims 13-22 and 38-47 are pending in this application.

### **Drawing Changes**

Accompanying this Response is a replacement page of the drawings on which Fig. 25A appears. In Fig. 25A, the spacing of the text in act 1010 has been corrected so that all of the text fits within the box for act 1010. No new matter has been added to the drawings.

### **Restriction Requirement**

Election of Group III, claims 13-22 and 38-46 is affirmed by Applicant.

### **35 U.S.C. § 103**

Claims 13-22 and 38-46 stand rejected under 35 U.S.C. §103(a) as being unpatentable over U.S. Patent Application Publication No. 2001/0056504 to Kuznetsov (hereinafter "Kuznetsov") in view of The Component Object Model Specification, Version 0.9 (hereinafter "Comspec"). Applicant respectfully submits that claims 13-22 and 38-46 are not obvious over Kuznetsov in view of Comspec.

Kuznetsov is directed to data exchange and data transfer (see, Title and ¶ 1). As discussed in the Abstract of Kuznetsov, Kuznetsov describes a high level transformation method and apparatus for converting data formats in the context of network applications, among other places. A flexible transformation mechanism is provided that facilitates generation of translation machine code on the fly. Kuznetsov discusses conversions between different XML formats (see, ¶ 6), as well as XML to HTML and WAP formats (see, ¶ 7).

Comspec is Version 0.9 of the Component Object Model Specification. As discussed in Comspec at §1.3, pp. 14-15, “The Component Software Solution: OLE’s COM”, the Component Object Model is an object-based programming model designed to promote software interoperability; that is, to allow two or more applications or “components” to easily cooperate with one another, even if they were written by different vendors at different times, in different programming languages, or if they are running on different machines running different operating systems. To support its interoperability features, COM defines and implements mechanisms that allow applications to connect to each other as *software objects*. A software object is a collection of related function (or intelligence) and the function’s (or intelligence’s) associated state. In other words, COM, like a traditional system service API, provides the operations through which a client of some service can connect to multiple providers of that service in a polymorphic fashion.

Furthermore, as discussed in Comspec at §1.3.2.2, p. 17, “COM’s Standards Enable Object Interoperability”, with COM, applications interact with each other and with the system through collections of function calls—also known

as methods or member functions or requests—called *interfaces*. An “interface” in the COM sense is a strongly typed *contract* between software components to provide a relatively small but useful set of semantically related operations. An interface is an articulation of an expected behavior and expected responsibilities, and the semantic relation of interfaces gives programmers and designers a concrete entity to use when referring to the contract.

Applicant respectfully submits that it would not have been obvious to one of ordinary skill in the art to combine Kuznetsov and Comspec. Kuznetsov, as discussed above, is directed to converting data formats. There is no discussion or mention in Kuznetsov of converting to anything other than simply other data formats, such as XML, HTML, or WAP formats. However, COM is much more than simply a new data format. For example, as discussed above COM includes the mechanisms that allow applications to connect to each other as software objects and interfaces that allow applications to interact with each other. An XML document does not have such mechanisms and interfaces, and thus cannot be simply converted into COM. There is no discussion or suggestion in Kuznetsov or Comspec of how such an XML document could be converted into COM because such mechanisms and interfaces would need to be generated and there is no discussion of how to generate such mechanisms and interfaces from XML. As COM is much more than a data format, and there is no discussion in Kuznetsov of converting data formats into anything other than other data formats, Applicant respectfully submits that it would not have been obvious to combine Kuznetsov and Comspec.

Furthermore, assuming for the sake of argument that Kuznetsov and Comspec were combined, Applicant respectfully submits that the combination does not disclose or suggest the elements of claim 13. Claim 13 recites:

One or more computer readable media having stored thereon a plurality of instructions that, when executed by a transformation engine, causes the transformation engine to:

access a plurality of constructs in an application programming interface description, wherein the description is written in an extensible markup language (XML) format; and

transform each of the plurality of constructs into code for a component object module (COM) application programming interface header file.

Applicant respectfully submits that there is no disclosure or suggestion in Kuznetsov or Comspec to transform each of a plurality of constructs into code for a COM application programming interface header file as recited in claim 13.

Kuznetsov, as discussed above, is directed to converting data formats. Even though Comspec discusses COM, there is no disclosure or suggestion of transforming a description in XML format into code for a COM application programming interface header file. Simply converting data formats (such as XML to XML or XML to HTML) does not provide any indication of transforming XML into code, much less code for a COM application programming interface header file. The mere existence of COM as described in Comspec does not provide any disclosure or suggestion of transforming XML into code, much less code for a COM application programming interface header file.

Accordingly, for at least these reasons, Applicant respectfully submits that claim 13 is allowable over Kuznetsov in view of Comspec.

Given that claims 14-22 depend from claim 13, Applicant respectfully submits that claims 14-22 are likewise allowable over Kuznetsov in view of Comspec for at least the reasons discussed above with respect to claim 13.

With respect to claim 38, Applicant respectfully submits that, as discussed above with respect to claim 13, it would not have been obvious to combine Kuznetsov and Comspec. Accordingly, for at least these reasons, Applicant respectfully submits that claim 38 is allowable over Kuznetsov in view of Comspec.

Given that claims 39-46 depend from claim 38, Applicant respectfully submits that claims 39-46 are likewise allowable over Kuznetsov in view of Comspec for at least the reasons discussed above with respect to claim 38.

Applicant respectfully requests that the §103 rejections be withdrawn.

### **New Claims**

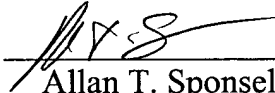
New claim 47 has been added. New claim 47 depends from claim 38, and Applicant respectfully submits that new claim 47 is allowable over the cited references at least because of its dependency on claim 38. Furthermore, Applicant respectfully submits that the cited references do not disclose or suggest a computer-readable medium as recited in claim 38, wherein a component object module (COM) application programming interface header file is to be generated for the application programming interface by transforming the data in the plurality of construct fields as recited in claim 47. For at least these reasons, Applicant respectfully submits that claim 47 is allowable over the cited references.

**Conclusion**

Claims 13-22 and 38-47 are in condition for allowance. Applicant respectfully requests reconsideration and issuance of the subject application. Should any matter in this case remain unresolved, the undersigned attorney respectfully requests a telephone conference with the Examiner to resolve any such outstanding matter.

Respectfully Submitted,

Date: 6/9/05

By:   
Allan T. Sponseller  
Reg. No. 38,318  
(509) 324-9256